USER'S GUIDE TO ADD NEW SURFACE VARIABLES OR NEW SURFACE ATTRIBUTES IN ARPEGE/IFS, ALADIN, AROME: CYCLE 32.

YESSAD K. (METEO-FRANCE/CNRM/GMAP/ALGO)

August 5, 2008

Contents

1 Introduction.

The new surface structure has been introduced by ECMWF (Mats Hamrud) in CY31R2, and now the surface buffers are split into a "SP" one (prognostic surface variables) and a "SD" one (diagnostic surface variables). The word "surface" can be sometimes misleading because we find not only surface or internal soil quantities, but also upper-air quantities at a specified level (temperature at 10 meters, height of the top of the PBL, total cloudiness). "SP" buffers are split into several groups, "SD" buffers also. They replace the old buffer GPPBUF which has been removed. The total number of fields reaches a total close to 200, and new fields are regularly added.

Adding new surface fields is not so easy that one could believe (even if the number of routines to be modified is rather small) and the purpose of the present paper is to provide a user's guide for people wanting to introduce new surface variables or new surface attributes. This is not a comprehensive documentation of all the surface features. Some other internal notes describe the new surface data flux.

Three types of "SD" surface variables can be found (independently of the different groups):

- The constants (for example the land-sea mask), which do not evolve and which are always input data in the physics.
- The diagnostics (for example the temperature at 10 meters), which are always output data in the physics: they are not conserved from one timestep to the following one.
- The pseudo-prognostic fields (like the Charnock constant), which are input of the physics, modified by the physics and the new value is conserved for the input of physics at the following timestep.

Additionally to that, some of them are read on a file; some of them can be post-processed; some of them can be assimilated.

Note that there are some other groups of "surface" fields which have not been put in the "SP" and "SD" buffers: for example some AROME surface fields which are still stored in GPARBUF, some VCLIU, VRADF, VTILE fields which are stored in specific buffers. Such quantities are out of the scope of this paper.

We first start to list the common features to be added when introducing surface fields (read or not on a file, post-processed or not, assimilated or not). Some details will be then given to describe:

- how to read/write new surface fields on files.
- how to post-process new surface fields.
- how to assimilate new surface fields.
- how to apply nudging on new surface fields.
- how to add new attributes.
- how to add a new class in SP or SD.

2 Add a new surface variable: main features.

2.1 No new group creation.

For the time being, the new variable is assumed to enter an existing group. To simplify, we assume that the new variable will have a generic code name NEWVAR.

∗ **List of routines to update:** For the time being we assume that we do not add any new attribute. There is a minimal list of routines which should be updated:

- module/surface_fields.F90
- setup/su_surf_flds.F90
- in some cases, phys_dmn/mf_phys.F90

∗ The first questions to answer:

- The first question to answer is to know if this is a prognostic variable (which must enter SP), or a diagnostic one (which must enter SD). If it must enter SD, is it a constant, a diagnostic or a pseudo-prognostic field?
- The second question is to determine in what group it must enter. For example, if this is a climatological constant or a geometric constant, it must enter SD, in the group VARSF.
- Is it a 3D field (several layers) or a 2D field (one layer)?

∗ **module/surface_fields.F90:** Now we assume that we have answered the previous questions. In SURFACE_FIELDS, we go in the topics “Group specific type definitions” and we add the attribute YNEWVAR in the adequate type definition (type TYPE_SFL_[group]). The attribute is itself a derivated-type variable (of type TYPE_SURF_MTL_2D or TYPE_SURF_MTL_3D according to the fact that this is a 2D variable or a 3D variable). A comment should always be provided at this place to know what is the variable NEWVAR.

∗ **setup/su_surf_flds.F90:** Go in the right group.

- The following line must be added:

```
YS[P or D]_[group]%YNEWVAR => YS[P or D]_[group]%Y[group](JPMAXSFLDS)
```

- After that, we find a list of instructions including a call to SETUP_SFLP2 or SETUP_SFLP3. At this stage, we should know when we need the field NEWVAR (for example for what type of physics). The cases LECMWF and .NOT.LECMWF are well separated. If, for METEO-FRANCE purpose, we need this field for LMPHYS=T, one should find:

```
IF (LMPHYS) THEN
  YS[P or D]_[group]%YNEWVAR => YS[P or D]_[group]%Y[group](YS[P or D]_[group]D%IPTR)
  CALL SETUP_SFLP2(YS[P or D]_[group]D,YS[P or D]_[group]%YNEWVAR,optional arguments)
ENDIF
```

  The optional arguments to be precised vary according to the field; for fields read on files the GRIB name or the ARPEGE file identifier appears; if KREQIN=1 that means that the field is read on a file.

- There is often a dimension (sometimes several dimensions) to be incremented at the beginning of the code corresponding to the group (for example IVSF in group VARSF). This dimension must be updated.

Note that, currently, there are no attribute LACTIVE in the definition of types TYPE_SURF_PTR_3D and TYPE_SURF_PTR_2D, to know if the surface field is needed and allocated (contrary to what exists for the GFL): that would be convenient to have such attribute for some specific applications.

∗ **phys_dmn/mf_phys.F90:** Some specific cautions should be taken for pseudo-historic “SD” fields, for purely diagnostic calls of MF_PHYS (this is the case for CLCONF(4)=’X’, sort of timestep zero in the XFU diagnostics). For such calls of MF_PHYS the initial content of surface variables should be saved in a local array, and restored at the end of MF_PHYS (I do not know if this question is relevant also for EC_PHYS).

2.2 New group creation.

The new variable is now assumed to enter a new group. To simplify, we assume that the new variable will have a generic code name NEWVAR, that the new group will have a generic code name NG (long version of the name: NGRUP), and that there will be one dimension INGRUP to increment.

∗ **List of routines to update:** For the time being we assume that we do not add any new attribute. There is a minimal list of routines which should be updated:

- module/surface_fields.F90
- setup/su_surf_flds.F90
- in some cases, phys_dmn/mf_phys.F90

∗ **The first questions to answer:**

- The first question to answer is to know if this is a prognostic variable (which must enter SP), or a diagnostic one (which must enter SD). If it must enter SD, is it a constant, a diagnostic or a pseudo-prognostic field?
- Is it a 3D field (several layers) or a 2D field (one layer)?

∗ **module/surface_fields.F90:** Now we assume that we have answered the previous questions.

- The new type TYPE_SFL_NGRUP must be defined (paragraph “Group specific type definitions”), for example (the new variable is assumed here to be a 2D one entering ’SD’):

```
! * Group NG=NGRUP: ??? (provide comments)
TYPE TYPE_SFL_[NGRUP]
TYPE(TYPE_SURF_MTL_2D),POINTER :: Y[NEWVAR]   ! ??? (provide comments)
TYPE(TYPE_SURF_MTL_2D),POINTER :: Y[NG](:)
END TYPE TYPE_SFL_[NGRUP]
```

- In the part “Data structures” (“Prognostic (multi time level) fields” if “SP”, or “one time level fields” if “SD”), add:

  ```
  ! NGRUP
  REAL(KIND=JPRB),ALLOCATABLE :: SD_[NG] (:,:,:)
  TYPE(TYPE_SURF_GEN)    :: YSD_[NG]D
  TYPE(TYPE_SFL_[NGRUP]) :: YSD_[NG]
  ```

- If the new class enters SP, routines GPPOPER and GPOPER should be updated; if the new class enters SD, routine GPOPER should be updated. GPOPER_2 must be called for 2D fields, GPOPER_3 for 3D fields.
- The new class must be added in routines ALLO_SURF and DEALLO_SURF.

∗ **setup/su_surf_flds.F90:** In SU_SURF_FLDS, we add the corresponding code for a new group in part 1 if this is a prognostic variable, in part 2 if this is a diagnostic variable. That provides a piece of code which looks like (the new variable is assumed here to be a 2D one entering ’SD’):

```
! NGRUP

INGRUP=0
IF(LECMWF) THEN
  LLACTIVE_NEWVAR=??? (when do we need NEWVAR?)
  IF (LLACTIVE_NEWVAR) INGRUP=INGRUP+1
ELSE
  LLACTIVE_NEWVAR=??? (when do we need NEWVAR?)
  IF (LLACTIVE_NEWVAR) INGRUP=INGRUP+1
ENDIF

IF(ASSOCIATED(YSD_[NG]%Y[NG])) DEALLOCATE(YSD_[NG]%Y[NG])
ALLOCATE(YSD_[NG]%Y[NG](JPMAXSFLDS))
CALL INI_SFLP2(YSD_[NG]D,YSD_[NG]%Y[NG],INGRUP,.FALSE.,'NGRUP - SD_[NG]    ')
YSD_[NG]%Y[NEWVAR]   => YSD_[NG]%Y[NG](JPMAXSFLDS)

IF(LECMWF) THEN
  IF (LLACTIVE_NEWVAR) THEN
    YSD_[NG]%Y[NEWVAR] => YSD_[NG]%Y[NG](YSD_[NG]D%IPTR)
    CALL SETUP_SFLP2(YSD_[NG]D,YSD_[NG]%Y[NEWVAR]  ,&
     & ... some optional arguments ...)
  ENDIF
ELSE
  IF (LLACTIVE_NEWVAR) THEN
    YSD_[NG]%Y[NEWVAR] => YSD_[NG]%Y[NG](YSD_[NG]D%IPTR)
    CALL SETUP_SFLP2(YSD_[NG]D,YSD_[NG]%Y[NEWVAR]  ,&
     & ... some optional arguments ...)
  ENDIF
ENDIF
```

About the optional arguments of SETUP_SFLP2, on can specify for example KGRIB (GRIB code), CDNAME (field name for the ARPEGE files), KTRAJ, KREQIN (KREQIN=1 means that the field must be read on a file). For a 3D variable, SETUP_SFLP3 must be called instead of SETUP_SFLP2.

∗ **phys_dmn/mf_phys.F90:** If the new variable is a pseudo-historic “SD” field, the same thing as for a new variable in a already existing class must be done.

3 Add a new surface variable: additional features when read/write this surface variable on ARPEGE files.

Note that there are no reference of surface variables to be read on an ARPEGE file in yomfa.F90, namfa.h, sufa.F90 which seem to be reserved for upper air fields (GMV, GFL) and surface GMVS fields. All the management previously done via module/yomphyds.F90, namelist/namphyds.h and setup/suphyds.F90 has been removed, so most of the setting-up done under SU_SURF_FLDS automatically provides the good setup to read/write on files (provided the optional argument KREQIN=1 is given). That ensures consistency between allocating surface fields and reading them (we don’t read a not allocated surface field in a file and we are sure that the name of the field

read/written on ARPEGE files is consistent with the default one given in SU_SURF_FLDS). The only features remaining in yomphyds.F90 and namphyds.h are for groups EXTRP, VEXTRA, VEXTR2.
But there is indeed some other routines to be updated if a new group is introduced. There is a minimal list of routines which should be updated:

- setup/sugridadm.F90
- setup/sugridg.F90
- dia/wrmlppadm.F90

∗ **setup/sugridadm.F90:** The new group must be introduced in part 3 (prognostic fields) or in part 4 (diagnostic fields) if the new group contains variables which should be read on ARPEGE files. Take the code for an existing group, adapt it for the new group (it must call RDGPFA and DISGRID).

∗ **setup/sugridg.F90:** Same type of update than in sugridadm.F90, but for surface fields which should be read on GRIB files.

∗ **dia/wrmlppadm.F90:** The new group must be introduced in part 3.2 (prognostic fields) or in part 3.3 (diagnostic fields) if the new group contains variables which should be written on ARPEGE files. Take the code for an existing group, adapt it for the new group (it must call RDGPFA and DISGRID).

∗ **Additional remarks:** In sugridadm.F90 and wrmlppadm.F90, reading/writing is currently done according the value of the attribute YS[P or D]_[group]%Y[group](:)%CNAME(:) (no reading/writing on files if this variable contains only blanks, reading/writing on files otherwise) and not according the value of the attribute YS[P or D]_[group]%Y[group](:)%NREQIN(:): it would be better in the future to use YS[P or D]_[group]%Y[group](:)%NREQIN(:).

4 Add a new surface variable: additional features when post-processing this surface variable.

Description of new features is not given in detail; this part must be described in a more comprehensive way in the future.

∗ **module/yomafn.F90 and namelist/namafn.h:** Variable GFP_[NEWVAR] must be added; it is desirable that the name into [NEWVAR] will be identical to the one used in the surface variables (avoid the discrepancies such as LSM in GFP_LSM and ITM in the surface data flux for land-sea mask!).

∗ **module/parfpos.F90:** If necessary, value of JPOSSGP must be increased.

∗ **other routines:** Adding a new variable GFP_[NEWVAR] will generate modifications in the following routines:

- pp_obs/fpiniphy.F90
- pp_obs/hpos.F90
- pp_obs/pregpfpos.F90 (part 1 must be updated if a new group is introduced).
- setup/suafn1.F90
- setup/suafn2.F90
- setup/suafn3.F90

and in some cases (generally climatic, vegetation or geometric constants) in the following routines:

- pp_obs/endpos.F90
- pp_obs/endvpos.F90
- pp_obs/fpcorphy.F90
- pp_obs/fpfillb.F90
- c9xx/rdecclimo.F90
- setup/sufpsuw.F90
- setup/sufptr2.F90
- setup/surfpds.F90

For ordering of the GFP..., if possible, keep the same order in YOMAFN, NAMAFN, SUAFN1, SUAFN2, SUAFN3.

More generally, look at all routines using YOMAFN to see if some physical surface post-processable variables are used, and if the new post-processable variable requires code updating.

5 Add a new surface variable: additional features when assimilating this surface variable.

This topic is not described in detail for the time being (it has to be written by people having a good knowledge into assimilating surface variables). That includes some CANARI routines.

6 Add a new surface variable: additional features when applying nudging on it.

Nudging is used for climatic simulations.

This topic is not described in detail for the time being (it has to be written by people having a good knowledge into nudging). Some of the variables XVU.. of YOMNUD contain reference values for the nudging of surface prognostic variables (surface moisture for example). When adding a new variable which can be nudged the question of updating the following routines must be answered:

- climate/updsst.F90 (and its call in adiab/cpg_gp.F90).
- dia/cpnudg.F90 (and its call in phys_dmn/mf_phys.F90).

A new XVU.. must be added in YOMNUD and all routines using the XVU.. must be updated: expect modifications in the following routines:

- dia/cpnudg.F90
- climate/updnuddm.F90
- climate/updsst.F90
- module/yomnud.F90

7 Add a new surface attribute.

We assume that we want to add a new attribute to the following data structures: TYPE_SURF_MTL_2D (resp. TYPE_SURF_MTL_3D): the generic name of this attribute is NEWATTR

∗ **List of routines to update:** There is a minimal list of routines which should be updated:

- module/surface_fields.F90
- setup/su_surf_flds.F90

∗ module/surface_fields.F90 and setup/su_surf_flds.F90:

- New attribute NEWATTR must be added in the definition of TYPE_SURF_MTL_2D (resp. TYPE_SURF_MTL_3D).
- Update INI_SFLP2 and SETUP_SFLP2 for 2D variables (resp. INI_SFLP3 and SETUP_SFLP3 for 3D variables).
- In some cases this new attribute may become a new optional dummy argument of SETUP_SFLP2 (2D) or SETUP_SFLP3 (3D). Update the calls to SETUP_SFLP2 and SETUP_SFLP3 when required in SU_SURF_FLDS.

8 Add a new surface field or group in climatologic files (configuration 923/E923, METEO-FRANCE purpose only).

To simplify, we assume that the new variable will have a generic code name NEWVAR. If a new group must be created for that, the new group will have a generic code name NG (long version of the name: NGRUP).

∗ **List of routines to update if the new variable enters an existing group:** Once known the group where the new variable must enter, the proper INCLI.. routine (INCLI2 to INCLI10) must be updated. Reading of the new surface field on a database, interpolating it, and writing it on an ARPEGE file, must be added in the proper INCLI..

∗ **List of routines to update if the new variable enters a new groupe:** It must be questioned if a new INCLI.. routine must be created or if the code relative to the new variable can enter an existing INCLI.. . In the first case, a new INCLI.. routine must be written, with the same structure as in INCLI2 to INCLI10 (Reading of the new surface field on a database, interpolating it, and writing it on an ARPEGE file). In the last case, see the previous paragraph.

∗ **Additional remarks:** New INCLI.. routines must be called by INCLI0. For each new variable it must be questioned if there is only an annual climatologic value or if there are different montly climatologic values.